

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **05233419 A**(43) Date of publication of application: **10.09.93**

(51) Int. CI

G06F 12/00(21) Application number: **04030445**(71) Applicant: **HITACHI LTD**(22) Date of filing: **18.02.92**(72) Inventor: **MORIMOTO MAYUMI
HAGA HIROHIDE**

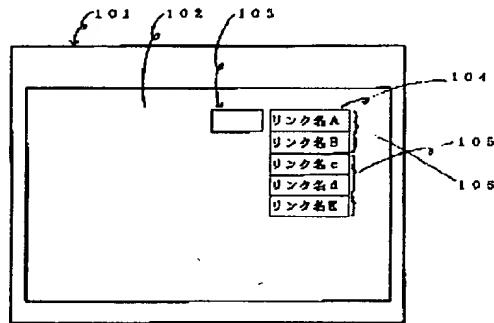
(54) HYPERTEXT SYSTEM

COPYRIGHT: (C)1993,JPO&Japio

(57) Abstract:

PURPOSE: To enable a user to freely define and alter link information without any danger of alteration by setting only link information defined by a user in link information as link information that the user can alter.

CONSTITUTION: This system is equipped with information 102 indicating the contents of a node 101 and an anchor 103 as an area where relative information to another node defined on the information 102 is defined. Further, the system has a list of link names defined in the anchor 103, the link names 105 of the link information that the user defines among the pieces of link information in the anchor 103, and link names 106 of link information that the system defines among the pieces of link information in the anchor 103. The link information previously defined by the system is not regarded as an object of link information that the user can alter. Consequently, the danger that the link information previously defined by the system is lost is eliminated and the user can freely define and alter the link information.



101
102
103
104
105
106

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平5-233419

(43)公開日 平成5年(1993)9月10日

(51)Int.Cl.⁵

G 06 F 12/00

識別記号 庁内整理番号

547 H 7232-5B

F I

技術表示箇所

審査請求 未請求 請求項の数4(全8頁)

(21)出願番号

特願平4-30445

(22)出願日

平成4年(1992)2月18日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 森本 ▲真▼由美

神奈川県川崎市麻生区王禅寺1099番地 株

式会社日立製作所システム開発研究所内

(72)発明者 芳賀 博英

神奈川県川崎市麻生区王禅寺1099番地 株

式会社日立製作所システム開発研究所内

(74)代理人 弁理士 小川 勝男

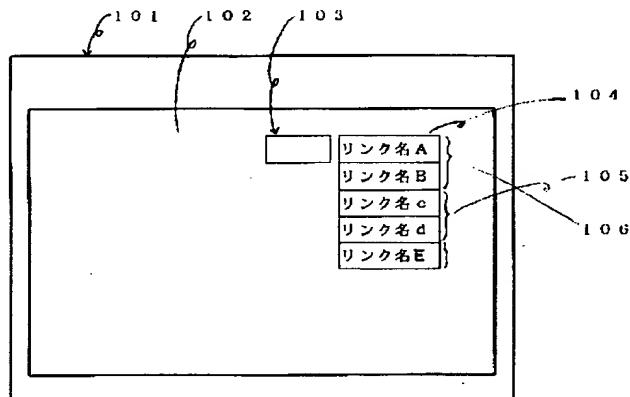
(54)【発明の名称】 ハイパーテキストシステム

(57)【要約】

【目的】ハイパーテキストシステムにおいて、システムが定義した変更不可能なリンク情報とユーザが定義した変更可能なリンク情報を混在させるとともに、システムが定義したリンク情報であるのか、ユーザが定義したリンク情報であるのかを、容易に識別できる方法を提供する。

【構成】個々のリンク情報に、システムが定義したリンク情報であるのか、ユーザが定義したリンク情報であるのかを示す識別子を付ける。

図1



【特許請求の範囲】

【請求項1】処理の対象となるデータと制御命令の列を表わすプログラムを格納するための少なくとも一つの読み書き可能な記憶装置と少なくとも一つの読み出し専用の記憶装置、前記記憶装置内のプログラムによって制御される中央処理装置、前記中央処理装置の処理の結果を表示する表示装置、前記表示装置上に表示されたデータに対して指示を与えるための入力装置、前記表示装置上の位置を指定するためのポインティングデバイスからなり、あるひとたまりの情報を複数の断片であるノードに分割し、各々の分割された断片情報の間の関連を示すために、リンク情報を定義することのできるハイパーテキストシステムにおいて、ハイパーテキストのノードとノードの関連を示すリンクとして、システムによってあらかじめ定義されていて、ユーザが変更不可能なリンク情報と、ユーザが定義できて、変更可能なリンク情報が混在することを特徴とするハイパーテキストシステム。

【請求項2】処理の対象となるデータと制御命令の列を表わすプログラムを格納するための少なくとも一つの読み書き可能な記憶装置、記憶装置内のプログラムによって制御される中央処理装置、前記中央処理装置の処理の結果を表示する表示装置、前記表示装置上に表示されたデータに対して指示を与えるための入力装置、前記表示装置上の位置を指定するためのポインティングデバイスからなり、あるひとたまりの情報を複数の断片であるノードに分割し、各々の分割された断片情報の間の関連を示すために、リンク情報を定義することのできるハイパーテキストシステムにおいて、ハイパーテキストのノードとノードの関連を示すリンクとして、システムによってあらかじめ定義されていて、ユーザが変更不可能なリンク情報と、ユーザが定義できて、変更可能なリンク情報を一つの読み書き可能な記憶装置上の別々のファイルに格納することを特徴とするハイパーテキストシステム。

【請求項3】処理の対象となるデータと制御命令の列を表わすプログラムを格納するための少なくとも一つの読み書き可能な記憶装置、前記記憶装置内のプログラムによって制御される中央処理装置、前記中央処理装置の処理の結果を表示する表示装置、前記表示装置上に表示されたデータに対して指示を与えるための入力装置、前記表示装置上の位置を指定するためのポインティングデバイスからなり、あるひとたまりの情報を複数の断片であるノードに分割し、各々の分割された断片情報の間の関連を示すために、リンク情報を定義することのできるハイパーテキストシステムにおいて、ハイパーテキストのノードと前記ノードの関連を示すリンクとして、システムによってあらかじめ定義されていて、ユーザが変更不可能なリンク情報と、前記ユーザが定義できて、変更可能なリンク情報を、前記リンク情報ごとに、前記ユーザが定義したリンク情報をあるのか、システムによって

あらかじめ定義された前記リンク情報であるのかを示す識別子を付けて、同一のファイルに格納することを特徴とするハイパーテキストシステム。

【請求項4】請求項1において、前記各々の断片情報に定義されている前記リンク情報の名称を表示装置上に表示する際に、システムによってあらかじめ定義されていて、ユーザが変更不可能なリンク情報と、前記ユーザが定義できて、変更可能なリンク情報の表現の形態を変えて表示するハイパーテキストシステム。

10 【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、ハイパーテキストを利用したシステムのリンク情報の定義方式に関する。

【0002】

【従来の技術】ハイパーテキストシステムは、あるひとたまりの情報を複数の断片情報であるノードに分割し、各々の分割された断片情報であるノードの間の関連を示すために、リンクで結ぶという共通の技術を実現している。ハイパーテキストの概念そのものについては、

20 「A S C I I」(株式会社アスキー、A S C I I 1 0 1 9 8 9 N o. 1 4 8) の p p. 2 2 1 ~ 2 3 6 に記載されている。また、具体的なハイパーテキストシステムの例は、「日経コンピュータ」別冊「ソフトウェア」(日経B P社、1 9 8 9 . 7 . 2 8) の p p. 1 1 1 ~ 2 0 8 に Hyper Card, 日本語Guide, 電脳机などの記述がある。

【0003】従来は、リンク情報として、全てシステムによってあらかじめ定義された固定的なリンク情報のみで、ユーザはリンク情報を定義することも変更することも不可能なシステムと、逆に、ユーザは自由にリンク情報を定義できて、システムによってあらかじめ定義されたリンク情報と、ユーザが定義したリンク情報の両方のリンク情報を変更することが可能なシステムの2種類があった。前者の例としては、C D - R O M等の書き換え不可能な記憶媒体に情報が格納するシステム、後者の例としてはHyper Cardがある。

【0004】我々が書籍を読むとき、書物の端にメモ書きをしたり、文章にアンダーラインを書き込んだりする。ハイパーテキストシステムでは、著者が与えた情報40 である文章、図表や脚注が、システムが与えた情報であり、この内の文章の内容を理解するために定義された図表や脚注が、システムによってあらかじめ定義されたリンク情報であるといえる。また、読者が書き込んだメモやアンダーラインは、ユーザが定義したリンク情報であるといえる。著者が与えた情報は、書き換えることができないが、読者が書き込んだメモやアンダーラインは、書き換えることができる。しかし、従来のハイパーテキストシステムには、書籍のように著者が与えた変更不可能である情報と読者が書き込んだ変更可能である情報50 が、混在しているものがいる。

【0005】また、読者が、著書にメモやアンダーラインを書くのと同様に、ユーザが、ノード間にリンク情報を定義するとき、自ら意図するものがあって、リンク情報を定義する。しかし、ユーザが、リンク情報からリンク先のノードへたどる場合、システムによってあらかじめ定義されたリンク情報なのか、ユーザが定義したリンク情報なのかが、わからない。これでは、ユーザが、リンク情報からリンク先のノードへたどるときに不便である。

【0006】

【発明が解決しようとする課題】従来のハイパーテキストシステムは、リンク情報の変更や定義ができないか、全てのリンク情報を変更の対象とするかのどちらかであり、書物のように変更不可能である情報と変更可能である情報が、混在していない。よって、システムによってあらかじめ定義された固定的なリンク情報のみからなるハイパーテキストシステムか、システムによってあらかじめ定義されたリンク情報を変更してしまう危険性はあるが、ユーザが自由にリンク情報を定義できるハイパーテキストシステムしかない。

【0007】また、ユーザがリンク情報を定義できる従来のハイパーテキストシステムでは、ユーザが、リンク情報をたどって別のノードに行くとき、システムによってあらかじめ定義されたリンク情報であるのか、ユーザが定義したリンク情報であるのかが、わからない。そのために、ユーザはあるノードにおいて、自ら意図して定義したリンク情報を、すぐに見つけ出すことができない場合がある。

【0008】本発明の目的は、システムによってあらかじめ定義されたリンク情報を、変更する危険性なしにユーザは自由にリンク情報を定義でき、変更できるハイパーテキストシステムを提供する。また、システムによってあらかじめ定義されたリンク情報であるのか、ユーザが定義したリンク情報であるのかを、識別できるように表示する方法を提供することである。

【0009】

【課題を解決するための手段】本発明は、ユーザは自由にリンク情報を定義できるが、ユーザが変更できるリンク情報の対象を、リンク情報のなかのユーザが定義したリンク情報のみとするハイパーテキストシステムを提供する。また、システムによってあらかじめ定義されたリンク情報とユーザが定義したリンク情報を、容易に識別できるように表現の形態を変えて表示する方式を提供する。

【0010】

【作用】システムによってあらかじめ定義されたリンク情報を、ユーザが変更可能なリンク情報の対象としないことによって、システムによってあらかじめ定義されたリンク情報を失う危険性なく、ユーザは自由にリンク情報を定義し、変更できる。リンク情報の表示の際に、

システムによってあらかじめ定義されたリンク情報とユーザが定義したリンク情報を、容易に識別できるように表現の形態を変えて表示することにより、ユーザインターフェースを向上させる。ユーザインターフェースの向上によって、ユーザは、システムによってあらかじめ定義されたリンク情報であるのか、ユーザが定義したリンク情報であるのかを、容易に識別でき、ユーザのたどりたいリンク情報を容易に見つけることができる。

【0011】また、このユーザインターフェースにおける

性能の向上によって、ユーザは、システムによってあらかじめ定義されたリンク情報であるのか、ユーザが定義したリンク情報であるのかを、容易に識別できるので、リンク情報に対する操作性の違い、例えば、変更可能であるリンク情報であるのか、変更不可能であるリンク情報であるのか等が、容易に識別できる。

【0012】

【実施例】図1は本発明を実施した場合に、表示装置上に表示される画面の一例を模式的に示したものである。

10 1はハイパーテキストのある一つのノードの情報を表示した領域、102はあるノード101の内容を示す情報、103はあるノード101の内容を示す情報102の上に定義された別のノードへの関連情報が定義されている領域であるアンカー、104はあるアンカー103に定義されたリンク名称の一覧、105はあるアンカー103に定義されたリンク情報のうちのユーザが定義したリンク情報のリンク名称、106はあるアンカー103に定義されたリンク情報のうちのシステムが定義したリンク情報のリンク名称である。

【0013】図2は本発明を実施するためのハードウェ

アの構成の一例である。201は中央処理装置、202はコマンド等を入力するためのキーボード等の入力装置、203は表示装置上の任意の位置を指示するためのマウス等のポインティングデバイス、204は表示装置、205は変更不可能な情報を格納している記憶装置、206は変更可能な情報を格納するための記憶装置、207は中央処理装置201の処理対象となる情報を一時的に記憶するための主記憶装置である。

【0014】図3はノードのリンク情報の主記憶装置上のデータ構造の一例である。301はある一つのノード

40 の情報、302はノードの上にあるアンカーの情報を保存している領域へのポインタ、303はノードの上にある一つのアンカーの情報、304はアンカーの始点の位置を示すための座標の情報、305はアンカーの終点の位置を示すための情報、306はアンカーに定義されているリンク情報を保存している領域へのポインタ、307はアンカーに定義されているある一つのリンク情報、308はリンク情報のリンク名称、309はあるリンク情報がシステムが定義したリンク情報であるのかユーザが定義したリンク情報であるのかを識別するための識別子、310はリンク情報からたどるためのリンク先のノ

ードへの情報を保存している領域へのポインタ、311はリンク先のノードへたどるための情報、312はリンク先のノードの情報である。一つのノードに複数のアンカーが定義されているときには、そのノードの上にある一つのアンカーの情報303が複数個定義されており、一つのアンカーに複数のリンク情報が定義されているときには、そのアンカーに定義されている一つのリンク情*

```

struct node {
    int *panchor; (1)
    :
} ; (2)

struct anchor {
    int sx, sy; (3)
    int fx, fy; (4)
    int *plink; (5)
    :
} (6)

struct link {
    char linkname[10]; (7)
    int flag; (8)
    int *preference; (9)
    :
} (10)

struct referent {
    char nodename[20]; (11)
    :
} (12)

```

(1) の構造体ノード(node)は、ノードの情報を保持しているテーブル 3 0 1 である。構造体ノードには、3 0 2 のアンカーの情報を保持している領域へのポインタ (2) がある。(3) の構造体アンカー(anchor)は、アンカーの情報を保持しているテーブル 3 0 3 である。構造体アンカーには、3 0 4 のアンカーの始点の位置を示す x, y 座標 (4), 3 0 5 のアンカーの終点の位置を示す x, y 座標、または、アンカーの幅と高さ (5), 3 0 6 のリンク情報を保持している領域へのポインタ (6) がある。(7) の構造体リンク(link)は、リンク情報を保持しているテーブル 3 0 7 である。構造体リンクには、3 0 8 のリンク情報の名称 (8), 3 0 9 の識別子 (9), 3 1 0 のリンク先のノードへの情報を保持している領域へのポインタ (10) がある。(11) の構造体リフェレント(referent)は、リンク先のノードへの情報を保持しているテーブル 3 1 1 である。構造体リフェレントには、3 1 2 のリンク先のノードの名称 (12) がある。

【0016】図4は図2で示したハードウェア構成のもとで、主記憶装置207に、図3のデータを作成する際の、本発明を実施するときのプログラムの処理を示す処理手順の一例である。以下では、図4に従って、処理手順の詳細を述べる。実現に当たっては、UNIXオペレ

*報307が複数個定義されている。305は、例えば、アンカーの終点の情報を示すための終点の座標でもよいし、アンカーの幅と高さの情報でもよい。312は、例えばリンク先のノードの名称を指定する。具体的には次のようなC言語のデータ構造で実現できる。

[0015]

(1)

(2)

(3)
(4)
(5)
(6)

(7)
(8)
(9)
1 0)

1 1)

1 2)

一タイミングシステム¹(以下UNIXと略記する)上で稼働しているH-Windowシステム(以下H ウィンドウと略記する)を例に取って述べる。

【0017】1:UNIXはUNIX System Laboratories, Inc.が開発し、ライセンスしています。

【0018】ノードの名称を、入力するなどの何らかの手段で指定されたある一つのノードを、表示装置204に表示するときに、そのノードの内部のデータを読み込む。まず、ステップ401で、外部記憶装置205から、システムが定義した表示するノードのデータを、主記憶装置207に読み込む。システムが定義したデータを読み込む際に、図3の識別子309に、システムが定義したデータであることを示す識別子を付加する。次に、ステップ402で、外部記憶装置206からユーザが定義した、表示するノードのデータを主記憶装置207に読み込む。ユーザが定義したデータを読み込む際に、図3の識別子309に、ユーザが定義したデータであることを示す識別子を付加する。システムが定義したデータとユーザが定義したデータを主記憶装置207に読み込んだら、ステップ403で、ノードを表示装置204に表示する。ノードが表示装置204の画面上に表示されたら、後は、そのノードが表示されている間、ユーザとの対話処理を行なえばよい。

【0019】この実施例では、システムが定義したデータとユーザが定義したデータを、異なる外部記憶装置に格納した場合を例に取って挙げたが、一つの読み書き可能な外部記憶装置に、別々のファイルとして格納した場合でも実現できる。一つの読み書き可能な外部記憶装置に格納する場合は、例えばファイル名称の拡張子で、システムが定義したデータファイルであるのか、ユーザが定義したデータファイルであるのかを区別することによって、データを読み込む際にどちらの情報であるのかを識別できるので、図4で作成したデータと同じデータが主記憶装置上に作成できる。また、一つのファイルに格納した場合でも実現できる。一つのファイルに格納する場合は、システムが定義したデータであるのか、ユーザ*

```

node(N 1)
anchor(N 1, A 1)
s_pos(A 1, s x, s y)
f_pos(A 1, f x, f y)
:
link(A 1, L 1)
:
referent(L 1, R 1)
:
```

N 1はノードの名称、A 1はアンカーの番号、s x, s yはアンカーの始点の座標、f x, f yはアンカーの終点の座標または幅と高さ、L 1はリンク名称、R 1はリンク先のノードの名称を意味している。(1)のノード()ではN 1というノードの定義を開始することの宣言、(2)のアンカー()ではアンカーの宣言、(3)のs_pos()ではアンカーの始点の宣言、(4)のf_pos()ではアンカーの終点の宣言、(5)のリンク()ではリンクの宣言、(6)のリフェレント()ではリンク先の宣言をしている。また、同一のファイルに格納したときは、(5)のリンクの宣言をリンク(A 1, L 1, Flag)に変更して、フラッグにシステムが定義したリンク情報であるのか、ユーザが定義したリンク情報であるのかを示す識別子をいれればよい。

【0022】図5は図2で示したハードウェア構成のもので、図3の主記憶処理装置207上のデータを用いて、本発明を実施するときのプログラムの処理を示す処理手順の一例である。以下では図5に従って処理手順の詳細を述べる。

【0023】ある一つのノードが、画面上に表示されている状態において、ステップ501でユーザがアンカーの選択を行なう。アンカーの選択は入力装置202のキーボード等を用いてもよいし、ポインティングデバイス203を用いてもよい。アンカーの選択を、入力装置202のキーボードを用いて行なうならば、ノード上で選択するアンカーの位置に、カーソルを移動させ、リターンキーを押下する。ポインティングデバイス203を用いて選択を行なうならば、選択する位置にポインティン

*が定義したデータであるのかを示す識別子を含めたデータを、ファイルに格納すればよい。異なるファイルに格納する場合も、システムが定義したデータであるのか、ユーザが定義したデータであるのかを示す識別子を含めたデータを、ファイルに格納してもよい。いずれの方法をとるにしても、ノードのデータを主記憶装置上に作成したとき、システムが定義したデータであるのか、ユーザが定義したデータであるのかを識別するための識別子があれば、本発明は実現できる。

10 【0020】外部記憶装置205や206には、具体的には次のようなデータを格納する。

【0021】

(1)

(2)

(3)

(4)

(5)

(6)

グデバイス203のカーソルを移動させて、クリックする。いずれの方法でアンカーを選択しても、選択した点がわかる。

【0024】以下では、ポインティングデバイス203を用いた選択法による手順を示す。例えばHウインドウでは、ポインティングデバイス203であるマウスを用いて選択した場合は、マウスからの情報が、エスケープシーケンスの形式で、ユーザプログラムに返される。このエスケープシーケンスには、マウスボタンのクリックされた座標データが含まれている。Hウインドウシステムのエスケープシーケンスの構成は、例えば「H I - U Xプログラマーズガイド ウィンドウ編」(2050-3-411-40)のpp. 198~200に記載されている。この機能を用いて、クリックした座標の値がわかる。

【0025】選択した座標の点が得られたら、次にステップ502で得られた点を含む部分に、アンカーがあるかどうかを調べる。これには、ユーザが選択した点のx, y座標が、共にアンカーの始点と終点の座標の間に入っているかどうかを、比較すればよい。あるアンカーが選択されたかどうかについては、そのアンカーの始点の座標と終点の座標がわかれればよい。ある一つのアンカーの位置についての情報は、テーブル303にある。テーブル303を検索して、アンカーの始点の情報304と終点の情報305から、アンカーの始点の座標と終点の座標が得られる。選択されたアンカーである場合には、ステップ503からステップ508を行ない、アンカーがない場合には、なにも行なわない。ステップ50

3では、選択されたアンカーに定義されているリンク情報へのポインタ306を認識する。

【0026】次に、ポインタ306に指定された情報領域のテーブル307から、主記憶装置209にリンク情報を読み出し、リンク名称を表示装置204に表示する。リンク名称を表示するための手順の一例を挙げる。リンク情報がなくなるまで、ステップ504からステップ508を繰り返すことによって、リンク名称を表示することができる。例えばテーブル307からステップ504で、一つ目のリンク情報のリンク名称308を読みだし、ステップ505とそのリンク情報の識別子309を読みだす。ステップ506で、リンク情報の識別子309が、システムが定義したリンク情報の識別子であるのか、ユーザが定義したリンク情報であるのかを判断し、システムが定義したリンク情報の識別子であれば、ステップ507で表示の形態をシステムが定義したリンク情報の形態で表示し、ユーザが定義したリンク情報であれば、ステップ508で表示の形態をユーザが定義したリンク情報の形態で表示する。表示の形態は、例えばシステムが定義したリンク情報の表示の形態をリバースした状態にし、ユーザが定義したリンク情報の表示の形態をノーマルな状態にするなどして、容易に識別できるような表示の形態に変えて表示する。ノーマルな状態の表示と、リバースの状態の表示を制御するためには、リンク名称を表示する前に、表示装置204に所定のエスケープシーケンスを送る。

【0027】Hウインドウシステムのエスケープシーケンスの構成は、例えば「H I - UXプログラマーズガイド ウィンドウ編」(2050-3-411-40)のpp.237~239に記載されている。これによって、表示の状態を変えることが実現できる。

【0028】以上の実施例においては、表示の形態の変化として、ノーマルな状態とリバースな状態を用いたが、例えばカラーを表示できる表示装置では、リンク情報の種類によって背景の色を変える等の方法で識別可能にしてもよい。

【0029】

【発明の効果】本発明は、システムによってあらかじめ

定義されたリンク情報を失う危険性なしに、ユーザは自由に各ノード間に自由にリンク情報を定義し、変更することができる。また、システムによってあらかじめ定義されたリンク情報と、ユーザが定義したリンク情報を、容易に識別できるように表現の形態を変えて表示することによって、ユーザインターフェースが向上し、ユーザは、システムによってあらかじめ定義されたリンク情報であるのか、ユーザが定義したリンク情報であるのかを容易に識別でき、ユーザのたどりたいリンク情報を容易に見つけることができるし、システムによってあらかじめ定義されたリンク情報であるのか、ユーザが定義したリンク情報であるのかを識別できるので、リンク情報に対する操作性の違い、例えば変更可能なリンク情報であるのか、変更不可能なリンク情報であるのか等が容易に識別できる。

【図面の簡単な説明】

【図1】本発明を実施した場合に表示装置上に表示される画面の説明図。

【図2】本発明を実施するためのハードウェアのプロック図。

【図3】ノードのリンク情報の主記憶装置上のデータ構造のブロック図。

【図4】本発明を実施するときのプログラムの処理を示す処理手順のフローチャート。

【図5】本発明を実施するときのプログラムの処理を示す処理手順のフローチャート。

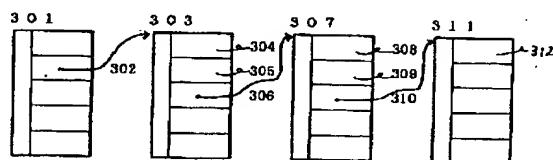
【符号の説明】

101…ハイパーテキストのある一つのノードの情報を表示した領域、102…あるノード101の内容を示す

30情報、103…あるノード101の内容を示す情報102の上に定義された別のノードへの関連情報が定義されている領域であるアンカー、104…あるアンカー103に定義されたリンク名称の一覧、105…あるアンカー103に定義されたリンク情報のうちのユーザが定義したリンク情報のリンク名称、106…あるアンカー103に定義されたリンク情報のうちのシステムが定義したリンク情報のリンク名称。

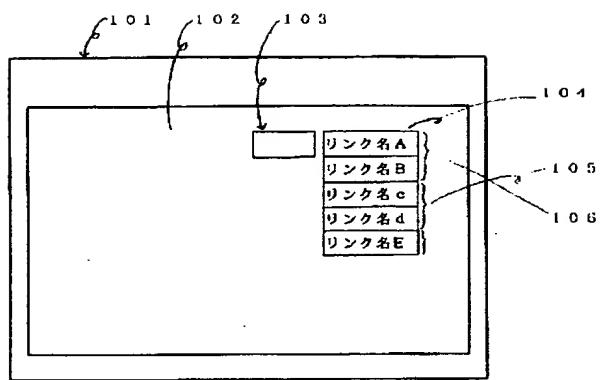
【図3】

図3



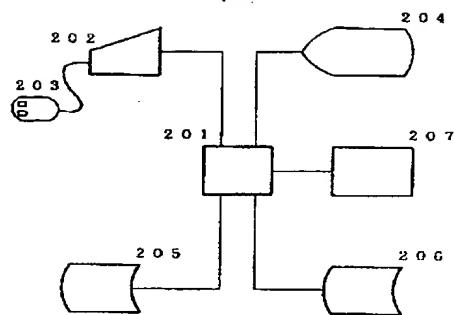
【図1】

図1



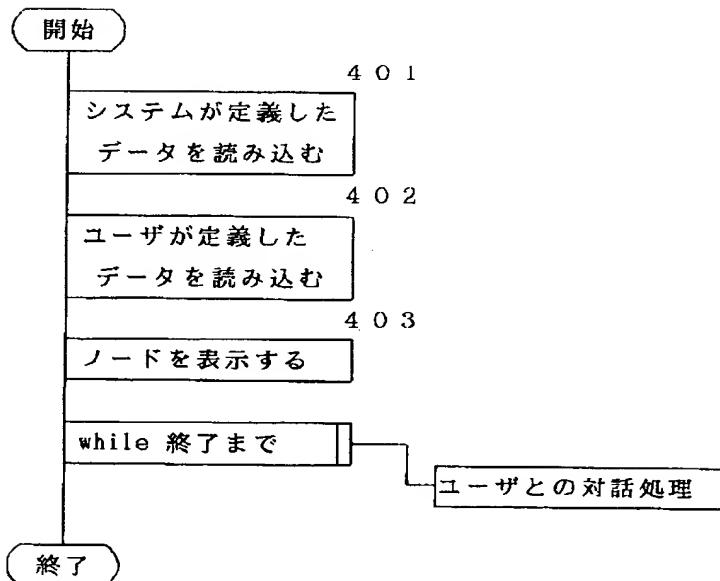
【図2】

図2



【図4】

図4



【図5】

